Introduction to Kubernetes

Glossary

- Kubernetes is a system for managing clusters of containers, including orchestration, scheduling, etc.
- Pods are the basic deployable units in a cluster. Pods have one or more tightly coupled containers.
- Services define abstractions across a logical set of Pods and a policy to access them.
- StatefulSet, DaemonSet and Deployment ensure that Pods are running at any given time with varying levels of guaranties and properties.
- Namespaces provide “virtual clusters”.

Pods

- Basic deployable units in Kubernetes. 1 Pod = 1..N Containers (usually 2/3)

- We can run multiple docker containers within a pod - imagine nginx server which serves files fetched from S3. First pod runs nginx container, second container fetches the file from S3 and puts it in a path available to nginx, then restarts nginx.

- Pods are defined using pod specifications written as YAML configuration and submitted to Kubernetes for scheduling via **kubectl**.

- **Demo** of scheduling a nginx web server Pod.

Service and Ingress

- Service - Expose inbound ports (TCP/UDP) on the Pod.
- Ingress - Collection of rules that allow inbound connections to reach the cluster Service(s) (e.g., name virtual hosts, fanouts).
- **Demo** of nginx Service and Ingress

Foobar websocket service on Kubernetes

- Go service packaged as foobar:1.0 docker image.
- Build and deploy the docker image.
- Deploy the service, ingress and deployment spec into Kubernetes.
- **Demo** of Docker build, Service, Ingress and Deployment

Further references

- Tutorial on katacoda: https://kubernetes.io/docs/tutorials/kubernetes-basics/
- Minikube: https://kubernetes.io/docs/getting-started-guides/minikube/
- Kelsey Hightower's presentation: https://www.youtube.com/watch?v=h8VnWL7WGu8
- Kubernetes API documentation: https://kubernetes.io/docs/api-reference/v1.7/
- Ingress examples: https://github.com/kubernetes/ingress/tree/master/examples